Melexis
Microelectronic Integrated Systems

Melexis USB LIN Master
Version 2.0 revision Fast Loader

Documentation
For Melexis USB LIN master used in demo setups and development kits

Table of Contents

1. General description

The Melexis USB LIN master can be used for all applications requiring a LIN master. It is designed to support the automatic configuration algorithms as they are defined by the LIN consortium. The Melexis Lin master is fully configurable by software through the USB port and supports all practical baud rates on the LIN bus.

This user manual describes the LIN master revision 2 (Fast loader). This LIN master is backwards compatible with LIN master revision 1.

2. Hardware

The LIN master consists of a microcontroller connected to the TH8061 physical layer interface of Melexis. It can be connected to the PC using the USB port. The LIN bus and power supply can be connected through connectors at one side of the device. There are 5 control LEDs:
1. PC connected: indicates that the PC is connected and the USB link is active.
2. PC communication: indicates that there is communication between the PC and the master.
3. M2S: indicates that a message with data field send by the master is travelling over the bus.
4. S2M: indicates that a message with data field send by a slave is travelling over the bus.
5. Vbus: indicates that the LIN bus is powered and that messages can be send.

The internal microcontroller and LEDs are powered by the USB bus. When the USB plug is connected, communication with the microcontroller over the USB line is possible. The LIN bus itself is not powered by the USB interface. Therefore an extra power supply has to be foreseen. The LIN bus voltage will be 1 diode voltage lower than the external power supply. As soon as the power supply is above 8V, the green LED will be on. When the LIN voltage is lower than 7V, the internal microcontroller will still send the requested LIN message to the physical layer. Because of the fact that the physical layer is not powered according to the LIN specifications, the signals on the LIN bus will not be valid. But the fact that the LIN voltage is out of spec does not block the functionality of the master. Even when the LIN voltage is out of spec, the microcontroller drives the physical interface of the LIN bus.

After connecting the USB cable, the microcontroller is powered. All 5 LEDs will be on for approximately 10 seconds. After those 10 seconds, some of the LEDs will flash and only the PC connected LED will be on. If the external power is connected, the green Vbus LED will be on too. From now on, all LEDs will react on PC and LIN messages.

Pinout of the different connectors:
- Red banana connector = battery voltage 12V typical.
- Black banana connector = ground
- Yellow banana connector = LIN bus

3. Software

This chapter describes the necessary steps to install the device drivers and all the software necessary to communicate with the LIN master.
All software is written for Windows XP Pro. It might be possible that is it working under other versions of windows, but this cannot be guaranteed.

3.1. Installation of the LIN master

This chapter describes the necessary installations to do to be able to work with the LIN master.
Note that **both software and device drivers installations are required** in order to work with the device and **must be done before the first time you connect the LIN Master** to the PC. Also, to be able to successfully accomplish these installations you need **administrative rights** on the PC.

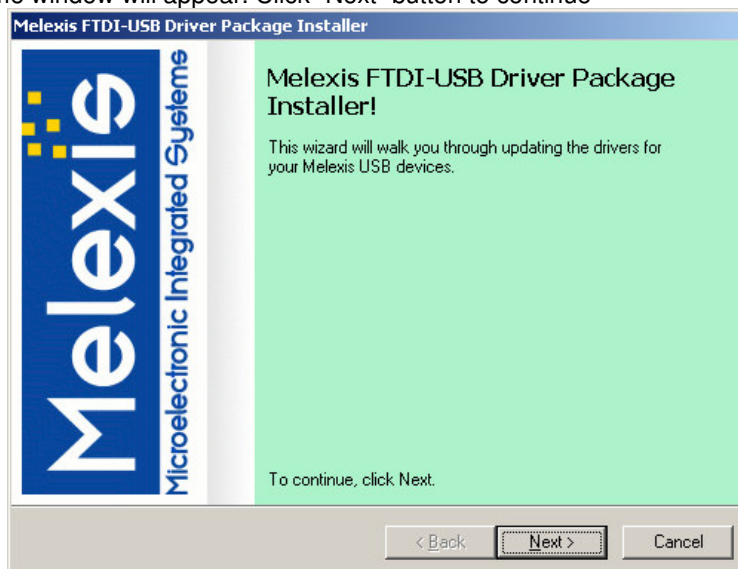### 3.1.1. Installing the USB driver

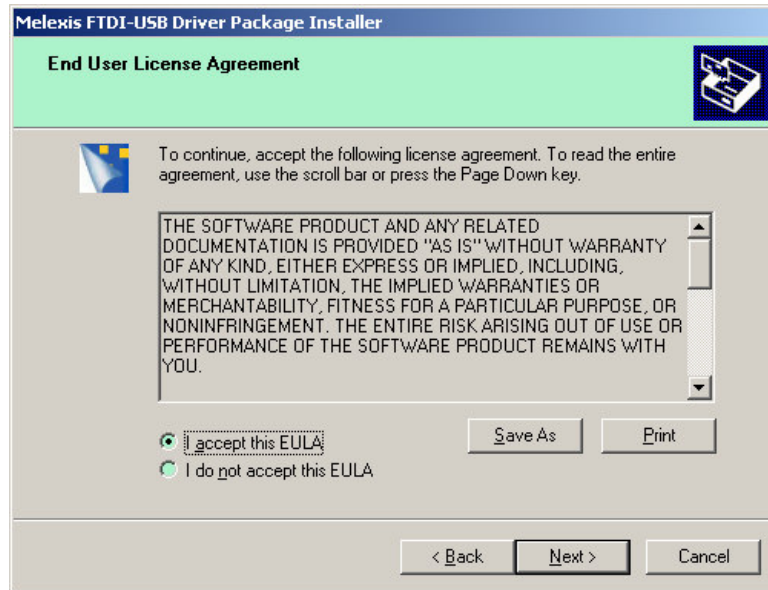Insert the CD into your PC
    1.   An autorun popup screen will appear
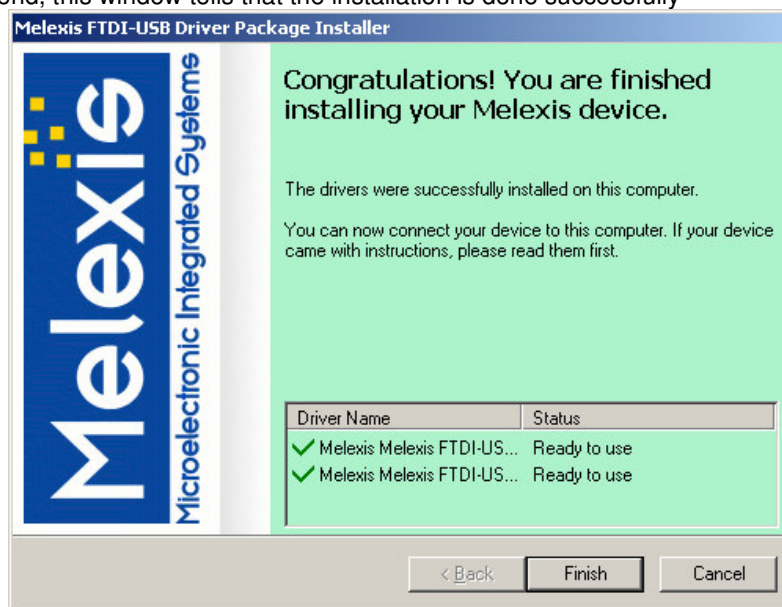

    2.   Click on the "Install USB driver" button to start the installation
    3.   A welcome window will appear. Click "Next" button to continue

    4.   You'll see a window with the license agreement and a few installation windows popping up. Accept the license and ignore all the windows.

5. At the end, this window tells that the installation is done successfully

The correct drivers are installed. From now on, the LIN master can be accessed via a virtual COM port. Note that the first time the device is connected to a particular USB port, an automatic virtual COM port installation is performed and a unique COM port number is assigned. Check the device manager of Windows XP to get the number of the COM port the LIN master is connected to (start->settings->control panel->system->hardware->device manager->Ports (COM & LPT)).

3.1.2. Installing the LIN master application software

After installing the USB drivers, the software to work with the LIN master have to be installed. This installation is done automatically.
Insert the CD into your PC
    1. An autorun popup screen will appear

    2. Click on the Install LinMaster button to start the installation
    3. You'll see a window with the license agreement and a few installation windows popping up. Accept the license and ignore all the windows.
    4. At the end, this window tells that the installation is done successfully
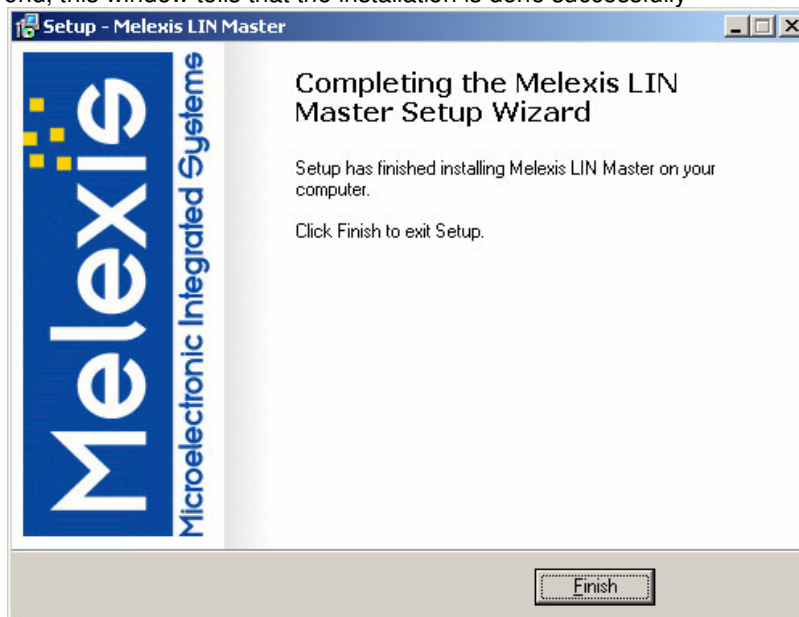
The LIN master is now ready to use.
LIN Master can be used through two applications – LIN commander which is described below and the user interface module for MPT application which is documented separately in LinMasterUI.pdf.
For those, who want to implement their own application, we provide a library along with documentation (LIN_PSF_Object_Model.pdf) and examples in LabView and MS Excel – all delivered on the CD.

3.2. LIN commander

The LIN master is delivered with a sample interface program that can be used to debug the bus when developing an application. This program is called the 'LIN commander' and a screenshot of it is displayed in Figure 1.

The configuration of the LIN messages is done by a LIN-file. This text file describes the LIN messages that can be sent over the bus. Hereafter, the syntax of such a LIN file is given together with an example:

- Comments start with an asterisk (*). The comment symbol should be placed in the beginning of a line. The whole line is ignored.
- Empty lines are ignored
- The END keyword indicated the end of the lin file. This keyword must be present, otherwise the file is ignored. Everything that is written after this end keyword is ignored.
- The complete syntax is case insensitive.
- The following keywords are defined:
    - *END*: indicate the end of the file
    - *LINBAUDRATE=*: defines the LIN baudrate on the bus. It should be written in the beginning of the file
    - *SCHEDULENAME=*: defines the name of the schedule.
    - *M2S*: defines a master to slave message, the master sends the header as well as the data bytes.
    - S2M: defines a slave to master message, the master sends the header of the message, and the slave the data bytes.
    - WKU: defines a wake up pulse.
    - CK13: defines a checksum according to LIN spec 1.3
    - CK20: defines the enhanced checksum of the LIN 2.0 version
- A description of a LIN message consists of two parts, separated by a colon (:)
    - name of the message: can be any possible string as long as it does not contain a colon.
    - body of the message.
- The body of the message consists out of 4 parts each separated by a comma (,)
    - message type:
        1. M2S for a master to slave message: the master sends the header, data bytes and checksum.
        2. S2M for a slave to master message: the master sends the header and the slave the data bytes and the checksum
        3. WKU for a wake up pulse. In case of a wake up pulse, not other parameters are necessary.
    - LIN ID: the LIN ID of the message, always send by the master
    - In case of a master to slave message, all the data bytes to be send by the master
      In case of a slave to master message, the number of data bytes that the slave should receive
    - A checksum type
        1. CK13 for checksum according to LIN spec. 1.3
        2. CK20 for an enhanced checksum according to LIN spec 2.0
- Multiple messages can be concatenated into one message string. Each message has to be separated by a semi colon. This can be done to send multiple messages in the same time slot.
- All numbers can be decimal or hexadecimal numbers. Hexadecimal numbers are preceded by a $ symbol.

```
*******************************************************************************
*              MELEXIS LIN DEBUG                                              *
*******************************************************************************
```

```
* Example
*---------
LINBAUDRATE = 19200
SCHEDULENAME = example
* a LIN sequence with one M2S LIN message
M2S               : M2S, $25, $aa, $23, $0a, $04, CK13;
* a LIN sequence with one S2M LIN message
S2M               : S2M, $26, $04;
* a LIN sequence grouping 2 LIN messages, one M2S and one S2M message.
M2S+S2M           : M2S, $25, $01, $02, $0a, $04, Ck20; S2M, $26, $04, CK13;
END
```
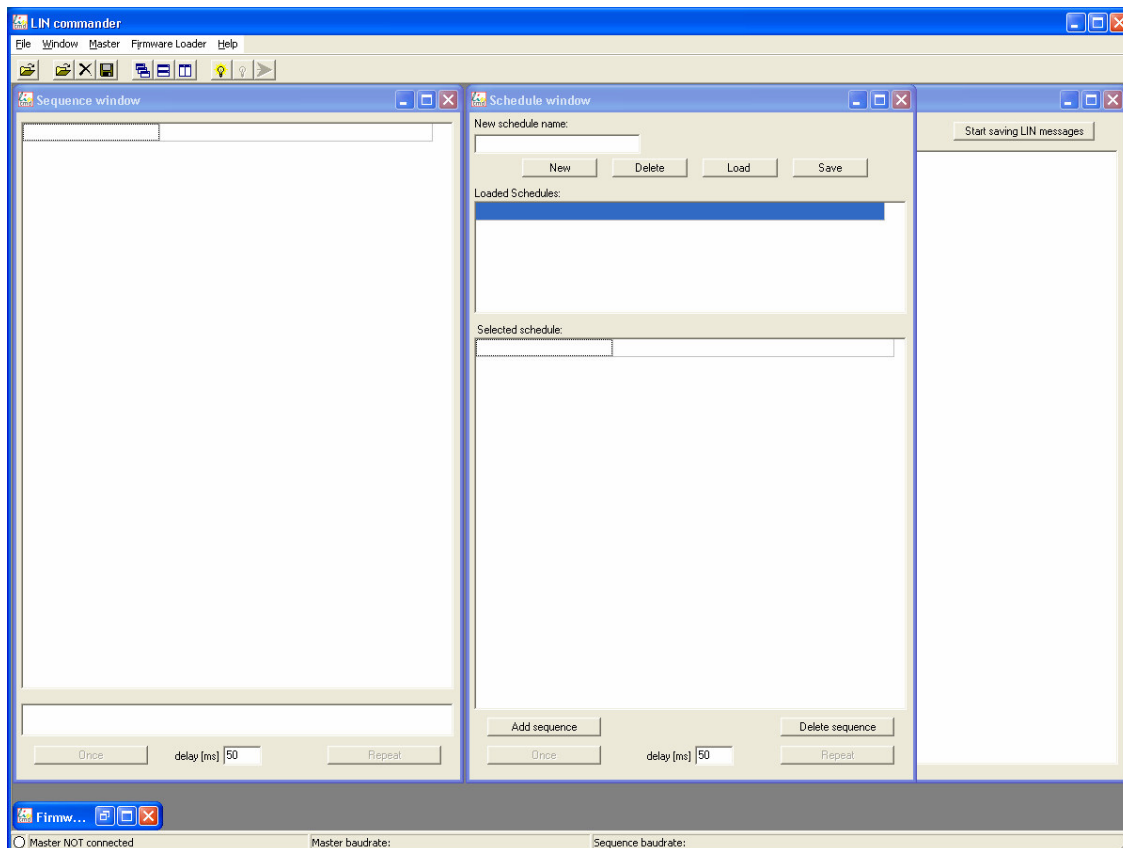

Figure 1: Screenshot of the LIN commander

1. Buttons to load the LIN files (schedules and sequences), (dis)connect the LIN master
2. The baud rate that is initialized in the LIN file and the real baudrate on the LIN bus
3. List of all configured LIN messages
4. The LIN message that will be send
5. The 'Once' and 'Repeat' buttons to send a message once or repeatedly
6. The time between 2 LIN messages when send repeatedly
7. LIN schedule and sequence windows
8. Message log window.

Mover over each button to see a small description on what the button does.

Take the following steps to work with the commander:
- Connect the LIN master via the USB link
- Connect a 12V power supply to the red and black banana connector (or the RJ45 cable).
- Double click the LIN commander icon on your desktop to start the software.

3.2.1. Lin Schedules and LIN Sequences

A LIN file can be interpreted as sequence or as schedule. The difference between them can be explained by an example.
In a system with multiple slaves, each slave has to react on a subset of the LIN messages on the bus. All possible messages on the bus can be described in a LIN sequence. The subset of messages per slave can be described in a schedule.
Lin Sequences are loaded in the sequence window, schedules in the schedule window. There can only be one loaded sequence at the time, while multiple opened schedules are possible. Each schedule has its name and all the schedules are listed in the 'Loaded schedules' list.

The syntax for sequences and schedules is the same and so schedules can be loaded into the sequence window and vice versa. The only difference between the two files is the SCHEDULENAME= keyword. If this is not present in the lin file that is loaded as schedule, the software automatically adds the name "anonymous" to the file when saving it.

New schedules can be assembled out of a sequence by writing the name of the new schedule in the 'New schedule name' field in the schedule window and clicking the 'new' button. The schedule content is displayed in the 'Selected Schedule' list. By clicking the 'Add Sequence' button the selected LIN message in the sequence is added to the schedule. Deleting a message from the schedule can be done by selecting the message in the schedule and clicking the 'delete sequence' button. Schedules can be saved and loaded by the 'Save' and 'Load' buttons. They are deleted from the schedule list by clicking the 'Delete' button. This does not delete the schedule from the hard disk.

Sending a LIN message in the sequence is done by selecting the message from the message list and clicking the 'Once' button. Sending it repeatedly is done by clicking the 'Repeat' button. The time interval is given in the 'Delay' box. The minimal practical delay is 10 ms due to the Windows timing system.

Sending a schedule is done by selecting it from the 'Loaded Schedules' list and clicking the 'Once' or 'Repeat' buttons. All the LIN messages of the schedule are sent. It is not possible to send only a part of the schedule.

3.2.2. Log window

The traffic over the LIN bus is logged in the Log window. All LIN messages are displayed here (message name, ID, data bytes and checksum). In some cases, extra info is given e.g no slave reaction on a S2M message. The messages can be logged in a csv file when needed.

3.2.3. Start working with the LIN commander

To start working with the LIN commander, do the following:
- Load a lin file by clicking the 'Load LIN file' button. The syntax of such an initialisation file is explained above
- Wait until the LIN master is detected by the software. If the software cannot detect the LIN master, a C++ exception is thrown. If this exception is thrown, close the software, disconnect the master, wait a few moments until the driver is unloaded from Windows and try again. This might happen the first time the Lin Master is connected after booting the PC.
- Select a message in the sequence list by clicking on it. The message will appear green in the box just above the 'Once' button.

- Click the Once button to sent the selected message once. The message (ID and databyte) will appear in the LIN message log window. If there are no databytes, only an ID (in cases that there is no slave that responds), only the ID is shown.
- Click the 'Repeat' button in case the message has to be send continuously. The time between two messages is displayed in the delay box. This is more an indication of the time between 2 messages. Every time a message has to be sent on the LIN bus, the PC has to give the command to the Lin Master. Therefore, a complete message has to travel through the USB stack on the PC. This might take some time, so the time indicated in the delay box is will be longer. In the test setup, the minimal time between 2 messages is 10ms.
  After clicking the repeat button, this button becomes a 'stop' button to stop sending the message.
- Clear the message log window by clicking the 'clear' button.
- It is also possible to save the logging into a csv file. Before starting the save, click the "start saving LIN messages" button and browse to the correct directory where you want to store the csv file.
- Because of the fact that the LIN master is a microcontroller with a LIN baudrate generating timer inside, it is not possible to have all imaginable baudrates. The real baudrate is indicated below the theoretical baudrate. A table of all possible baudrates is given in Table 1.
- Select the 'view logging' selection box to show the Lin message log window.
- Click the disconnect button to disconnect the LIN master. The application does not close. The software can be reactivated by clicking the 'connect' button.

3.2.4. Reprogramming of Melexis modules using the LIN Master

Some of the Melexis microcontroller based devices can be reprogrammed using the Melexis LIN master. Therefore, the 'Firmware Loader' command is developed.

If your project has this function, refer to your Melexis contacts on how to use these functionalities.

For designers who want to build their own PC application driving the LIN master, it is possible to get the source code for a framework to use the library and LIN master. This source code is developed with Borland C++ builder 5.0. Refer to you Melexis contact to get these libraries.

4. Appendix A: Possible LIN baudrates

Byte 2	Baudrate	Byte 2	Baudrate	Byte 2	Baudrate	Byte 2	Baudrate	Byte 2	Baudrate
1.00	125.00	53.00	4.63	105.00	2.36	157.00	1.58	209.00	1.19
2.00	83.33	54.00	4.55	106.00	2.34	158.00	1.57	210.00	1.18
3.00	62.50	55.00	4.46	107.00	2.31	159.00	1.56	211.00	1.18
4.00	50.00	56.00	4.39	108.00	2.29	160.00	1.55	212.00	1.17
5.00	41.67	57.00	4.31	109.00	2.27	161.00	1.54	213.00	1.17
6.00	35.71	58.00	4.24	110.00	2.25	162.00	1.53	214.00	1.16
7.00	31.25	59.00	4.17	111.00	2.23	163.00	1.52	215.00	1.16
8.00	27.78	60.00	4.10	112.00	2.21	164.00	1.52	216.00	1.15
9.00	25.00	61.00	4.03	113.00	2.19	165.00	1.51	217.00	1.15
10.00	22.73	62.00	3.97	114.00	2.17	166.00	1.50	218.00	1.14
11.00	20.83	63.00	3.91	115.00	2.16	167.00	1.49	219.00	1.14
12.00	19.23	64.00	3.85	116.00	2.14	168.00	1.48	220.00	1.13
13.00	17.86	65.00	3.79	117.00	2.12	169.00	1.47	221.00	1.13
14.00	16.67	66.00	3.73	118.00	2.10	170.00	1.46	222.00	1.12
15.00	15.63	67.00	3.68	119.00	2.08	171.00	1.45	223.00	1.12
16.00	14.71	68.00	3.62	120.00	2.07	172.00	1.45	224.00	1.11
17.00	13.89	69.00	3.57	121.00	2.05	173.00	1.44	225.00	1.11
18.00	13.16	70.00	3.52	122.00	2.03	174.00	1.43	226.00	1.10
19.00	12.50	71.00	3.47	123.00	2.02	175.00	1.42	227.00	1.10
20.00	11.90	72.00	3.42	124.00	2.00	176.00	1.41	228.00	1.09
21.00	11.36	73.00	3.38	125.00	1.98	177.00	1.40	229.00	1.09
22.00	10.87	74.00	3.33	126.00	1.97	178.00	1.40	230.00	1.08
23.00	10.42	75.00	3.29	127.00	1.95	179.00	1.39	231.00	1.08
24.00	10.00	76.00	3.25	128.00	1.94	180.00	1.38	232.00	1.07
25.00	9.62	77.00	3.21	129.00	1.92	181.00	1.37	233.00	1.07
26.00	9.26	78.00	3.16	130.00	1.91	182.00	1.37	234.00	1.06
27.00	8.93	79.00	3.13	131.00	1.89	183.00	1.36	235.00	1.06
28.00	8.62	80.00	3.09	132.00	1.88	184.00	1.35	236.00	1.05
29.00	8.33	81.00	3.05	133.00	1.87	185.00	1.34	237.00	1.05
30.00	8.06	82.00	3.01	134.00	1.85	186.00	1.34	238.00	1.05
31.00	7.81	83.00	2.98	135.00	1.84	187.00	1.33	239.00	1.04
32.00	7.58	84.00	2.94	136.00	1.82	188.00	1.32	240.00	1.04
33.00	7.35	85.00	2.91	137.00	1.81	189.00	1.32	241.00	1.03
34.00	7.14	86.00	2.87	138.00	1.80	190.00	1.31	242.00	1.03
35.00	6.94	87.00	2.84	139.00	1.79	191.00	1.30	243.00	1.02
36.00	6.76	88.00	2.81	140.00	1.77	192.00	1.30	244.00	1.02
37.00	6.58	89.00	2.78	141.00	1.76	193.00	1.29	245.00	1.02
38.00	6.41	90.00	2.75	142.00	1.75	194.00	1.28	246.00	1.01
39.00	6.25	91.00	2.72	143.00	1.74	195.00	1.28	247.00	1.01
40.00	6.10	92.00	2.69	144.00	1.72	196.00	1.27	248.00	1.00
41.00	5.95	93.00	2.66	145.00	1.71	197.00	1.26	249.00	1.00
42.00	5.81	94.00	2.63	146.00	1.70	198.00	1.26	250.00	1.00
43.00	5.68	95.00	2.60	147.00	1.69	199.00	1.25	251.00	0.99

Application note

Rev. 004

Oct/06

44.00	5.56	96.00	2.58	148.00	1.68	200.00	1.24	252.00	0.99
45.00	5.43	97.00	2.55	149.00	1.67	201.00	1.24	253.00	0.98
46.00	5.32	98.00	2.53	150.00	1.66	202.00	1.23	254.00	0.98
47.00	5.21	99.00	2.50	151.00	1.64	203.00	1.23	255.00	0.98
48.00	5.10	100.00	2.48	152.00	1.63	204.00	1.22		
49.00	5.00	101.00	2.45	153.00	1.62	205.00	1.21		
50.00	4.90	102.00	2.43	154.00	1.61	206.00	1.21		
51.00	4.81	103.00	2.40	155.00	1.60	207.00	1.20		
52.00	4.72	104.00	2.38	156.00	1.59	208.00	1.20		

Table 1: Possible LIN baudrates

Application note

Rev. 004

Oct/06

5. References

1) Lin specification package version 1.3 (see http://www.lin-subbus.org/)

6. Disclaimer

Devices sold by Melexis are covered by the warranty and patent indemnification provisions appearing in its Term of Sale. Melexis makes no warranty, express, statutory, implied, or by description regarding the information set forth herein or regarding the freedom of the described devices from patent infringement. Melexis reserves the right to change specifications and prices at any time and without notice. Therefore, prior to designing this product into a system, it is necessary to check with Melexis for current information. This product is intended for use in normal commercial applications. Applications requiring extended temperature range, unusual environmental requirements, or high reliability applications, such as military, medical life-support or life-sustaining equipment are specifically not recommended without additional processing by Melexis for each application.
The information furnished by Melexis is believed to be correct and accurate. However, Melexis shall not be liable to recipient or any third party for any damages, including but not limited to personal injury, property damage, loss of profits, loss of use, interrupt of business or indirect, special incidental or consequential damages, of any kind, in connection with or arising out of the furnishing, performance or use of the technical data herein. No obligation or liability to recipient or any third party shall arise or flow out of Melexis' rendering of technical or other services.
© 2002 Melexis NV. All rights reserved.

For the latest version of this document, go to our website at:

www.melexis.com

Or for additional information contact Melexis Direct:

Europe and Japan:
Phone: +32 13 67 04 95
E-mail: sales_europe@melexis.com

All other locations:
Phone: +1 603 223 2362
E-mail: sales_usa@melexis.com

QS9000, VDA6.1 and ISO14001 Certified